

## CLAIMS

We claim:

Sub 1817 1 ~~A method of restructuring a program comprising basic blocks for execution by a processor~~  
2 ~~having a memory hierarchy comprising a plurality of levels, said method comprising the steps~~  
3 ~~of:~~

4 ~~a) constructing a Program Execution Graph (PEG) from control flow and~~  
5 ~~frequency information from a profile of the program, the PEG comprising a weighted~~  
6 ~~undirected graph comprising nodes representing the basic blocks and edges representing~~  
7 ~~transfer of control between pairs of the basic blocks, each of the nodes having a weight equal~~  
8 ~~to the size of the basic block represented by the node, each of the edges having a weight~~  
9 ~~equal to a frequency of transition between a pair of basic blocks represented by a pair of nodes~~  
10 ~~connected by the edge;~~

11 ~~b) partitioning the nodes of the PEG into clusters such that a sum of weights of~~  
12 ~~the edges whose endpoints are in different clusters is minimized, and such that for any cluster,~~  
13 ~~a sum of weights of the nodes in the cluster is no greater than an upper bound; and~~

14 ~~c) restructuring the basic blocks into contiguous code corresponding to the~~  
15 ~~clusters.~~

1 2. The method of claim 1 further comprising the steps of:

2 d) constructing a next PEG from the clusters of the partitioned PEG such that a  
3 node in the next PEG corresponds to a cluster in the partitioned PEG, and such that there is  
4 an edge between two nodes in the next PEG if there is an edge between components of the  
5 clusters represented by the two nodes; and

6 e) assigning a weight to each node of the next PEG; and

7 f) assigning a weight to an edge between a pair of nodes of the next PEG  
8 representing a pair of clusters of the partitioned PEG, the edge weight being a summation of  
9 weights of edges in the partitioned PEG having endpoints in the pair of clusters in the  
10 partitioned PEG.

1 3. The method of claim 2 further comprising the step of:

2 f) repeating steps b through f.

1 4. The method of claim 1 wherein the upper bound is a multiple of a size of a level of the  
2 memory hierarchy.

1 5. The method of claim 3 wherein the upper bound for a level of the memory hierarchy, other  
2 than a first level, is a size of the memory hierarchy level divided by an upper bound used to  
3 partition a next lower level of the memory hierarchy.

1 6. The method of claim 3 further comprising the steps of:

2 removing a basic block whose size is greater than the upper bound from the  
3 partitioning step; and

4 reintegrating the basic block whose size is greater than the upper bound into a next  
5 repetition of steps b through f.

2025 RELEASE UNDER E.O. 14176

1 7. An article of manufacture for use in a computer system for restructuring a program  
2 comprising basic blocks for execution by a processor having a memory hierarchy comprising a  
3 plurality of levels, said article of manufacture comprising a computer-readable storage medium  
4 having a computer program embodied in said medium which may cause the computer system  
5 to:

6 a) construct a Program Execution Graph (PEG) from control flow and frequency  
7 information from a profile of the program, the PEG comprising a weighted undirected graph  
8 comprising nodes representing the basic blocks and edges representing transfer of control  
9 between pairs of the basic blocks, each of the nodes having a weight equal to the size of the  
10 basic block represented by the node, each of the edges having a weight equal to a frequency  
11 of transition between a pair of basic blocks represented by a pair of nodes connected by the  
12 edge;

13 b) partition the nodes of the PEG into clusters such that a sum of weights of the  
14 edges whose endpoints are in different clusters is minimized, and such that for any cluster, a  
15 sum of weights of the nodes in the cluster is no greater than an upper bound, and

16 c) restructure the basic blocks into contiguous code corresponding to the clusters.

1 8. The article of manufacture of claim 7 wherein the computer program may further cause the  
2 computer system to:

3 d) construct a next PEG from the clusters of the partitioned PEG such that a node  
4 in the next PEG corresponds to a cluster in the partitioned PEG, and such that there is an  
5 edge between two nodes in the next PEG if there is an edge between components of the  
6 clusters represented by the two nodes; and

7 e) assign a weight to each node of the next PEG; and

8 f) assign a weight to an edge between a pair of nodes of the next PEG  
9 representing a pair of clusters of the partitioned PEG, the edge weight being a summation of  
10 weights of edges in the partitioned PEG having endpoints in the pair of clusters in the  
11 partitioned PEG.

1 9. The article of manufacture of claim 8 wherein the computer program may further cause the  
2 computer system to:

3 ~~e) repeat steps b through f.~~

1 10. The article of manufacture of claim 7 wherein the upper bound is a multiple of a size of a level  
2 of the memory hierarchy.

1 11. ~~The article of manufacture of claim 10 wherein the upper bound for a level of the memory~~  
2 ~~hierarchy, other than a first level, is a size of the memory hierarchy level divided by an upper~~  
3 ~~bound used to partition a next lower level of the memory hierarchy.~~



1 13. A computer system for restructuring a program comprising basic blocks for execution by a  
2 processor having a memory hierarchy comprising a plurality of levels, said computer system  
3 comprising:

4 a) a Program Execution Graph (PEG) constructed from control flow and  
5 frequency information from a profile of the program, the PEG comprising a weighted  
6 undirected graph comprising nodes representing the basic blocks and edges representing  
7 transfer of control between pairs of the basic blocks, each of the nodes having a weight equal  
8 to the size of the basic block represented by the node, each of the edges having a weight  
9 equal to a frequency of transition between a pair of basic blocks represented by a pair of nodes  
10 connected by the edge;

11 b) a partition of the nodes of the PEG into clusters such that a sum of weights of  
12 the edges whose endpoints are in different clusters is minimized, and such that for any cluster,  
13 a sum of weights of the nodes in the cluster is no greater than an upper bound; and

14 c) a restructuring of the basic blocks into contiguous code corresponding to the  
15 clusters.

1 14. The computer system of claim 13 further comprising:

2 d) a next PEG constructed from the clusters of the partitioned PEG such that a  
3 node in the next PEG corresponds to a cluster in the partitioned PEG, and such that there is  
4 an edge between two nodes in the next PEG if there is an edge between components of the  
5 clusters represented by the two nodes; and

6 e) a weight assigned to each node of the next PEG; and

7 f) a weight assigned to an edge between a pair of nodes of the next PEG  
8 representing a pair of clusters of the partitioned PEG, the edge weight being a summation of  
9 weights of edges in the partitioned PEG having endpoints in the pair of clusters in the  
10 partitioned PEG.

1 15. The computer system of claim 14 further comprising:

2 f) a repetition of elements b through f.

1 16. The computer system of claim 13 wherein the upper bound is a multiple of a size of a level of  
2 the memory hierarchy.

1 17. The computer system of claim 15 wherein the upper bound for a level of the memory  
2 hierarchy, other than a first level, is a size of the memory hierarchy level divided by an upper  
3 bound used to partition a next lower level of the memory hierarchy.



~~18. The computer system of claim 15 further comprising:~~

a removal of a basic block whose size is greater than the upper bound from the partitioning step; and

a reintegration of the basic block whose size is greater than the upper bound into a next repetition of elements b through f.